SPIDER

Elemental Analysis in the SPIDER Project

Jeffrey Maddalon
NASA Langley Research Center
j.m.maddalon@nasa.gov

FAA/NASA Software and CEH Conference
Norfolk, Virginia
July 27, 2004

The SPIDER Project

- Scaleable Processor-Independent Design for Enhanced Reliability (SPIDER)
- Project Goals
  - Develop case study application of DO-254
  - Demonstrate application of formal methods in certification context
  - Develop research platform for exploring recovery from correlated transient faults

SPIDER

- A family of real-time, embedded, fault-tolerant Integrated Modular Avionics (IMA) architectures
- Targeted for critical avionics functions such as flight control.
- A complete fault-tolerance solution
  - Communication subsystem (ROBUS)
  - Low-level interfacing software
  - Application interface in the form of fault-tolerant middleware

Verification Team

Elemental Analysis

- Wilfredo Torres-Pomales
- Paul Miner
- Mahyar Malekpour
- Jeff Maddalon

Formal Methods

- Paul Miner
- Alfons Geser
- Lee Pike
- Radu Siminiceanu
- Jeff Maddalon

DO-254 Design Assurance

- For level A & B functions, appendix B of DO-254 describes
  - Functional Failure Path Analysis as a way to develop a design assurance strategy and
  - Specific design assurance methods
    - Including advanced verification methods like Elemental Analysis

DO-254 Design Assurance Methods

- Architectural Mitigation
- Service History
- Advanced Verification Methods
  - Elemental Analysis
  - Safety-Specific Analysis
  - Formal Methods

Applicant may propose additional methods

DO-254 Design Assurance Methods

- Architectural Mitigation
- ~~Service History~~ N/A because design is new
- Advanced Verification Methods
  - Elemental Analysis
  - ~~Safety-Specific Analysis~~ N/A because design is independent of aircraft function
  - Formal Methods

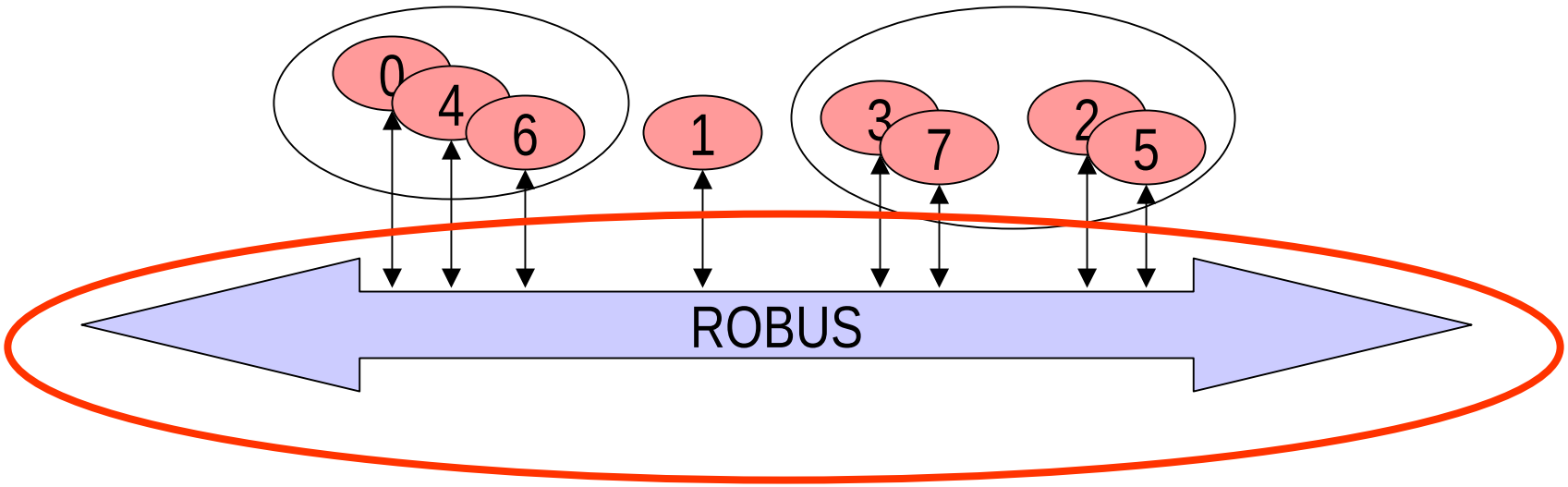
Applicant may propose additional methods

In this talk…

- Functional Failure Path Analysis
- Architectural Mitigation
- Elemental Analysis
- Formal Methods

- No COTS considerations
- Only part of the design is analyzed
  - "Input Unit" of "ROBUS Protocol Processor"

Sample SPIDER Configuration



ROBUS (Reliable Optical Bus)

- ROBUS is SPIDER's communication subsystem
- Contains no software
- Three types of nodes:
  - Processing Elements (PE)
  - Bus Interface Unit (BIU)
  - Redundancy Management Unit (RMU)
- Primary Functions
  - Message Broadcast
  - Time Reference
  - Self-diagnosis
  - Communication Schedule Update

Failures contained by ROBUS

ROBUS must tolerate

- A bounded number of internal physical failures
- Arbitrary failure in any attached PE
  - physical or design
  - hardware or software
- Cannot tolerate a design error within ROBUS

How to achieve these?

- ROBUS Architecture
- Markov analysis calculates Pr(enough good hardware)
- "Overlapping combinations" of design assurance methods (elemental and formal) provide

  enough good hardware => correct operation

Functional Failure Path Analysis

- A SPIDER system can host different applications on different PEs
  - These PEs can have different design assurance levels
- A misbehaving PE cannot be allowed to interfere with the communication of other PEs
  - If PEs could interfere with ROBUS communication, then *any function* could be compromised
- ROBUS provides “robust partitioning” of communication
  - per SC-200 (Modular Avionics)

Architectural Mitigation

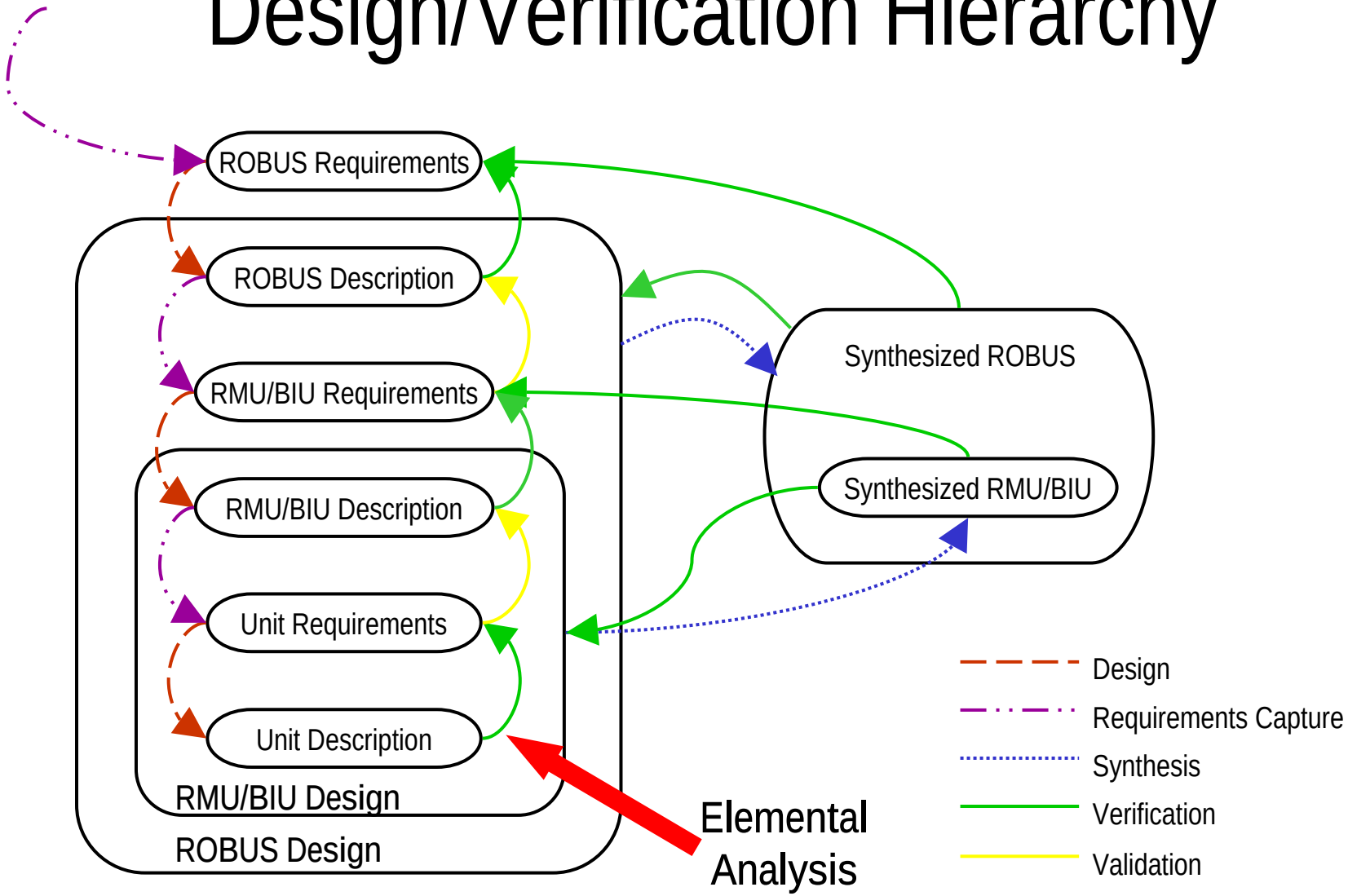
- Uses architectural features such as dissimilar implementations, redundancy, monitors, etc. to mitigate design and implementation errors.
- Design assurance of ROBUS does not use architectural mitigation
- However, the design of ROBUS allows architectural mitigation at the system level
  - ROBUS is designed to mitigate *arbitrary* PE faults
  - PEs are not required to use the same hardware

DO-254 Advanced Verification

- Page B-1 describes “overlapping, layered combinations” of design assurance methods

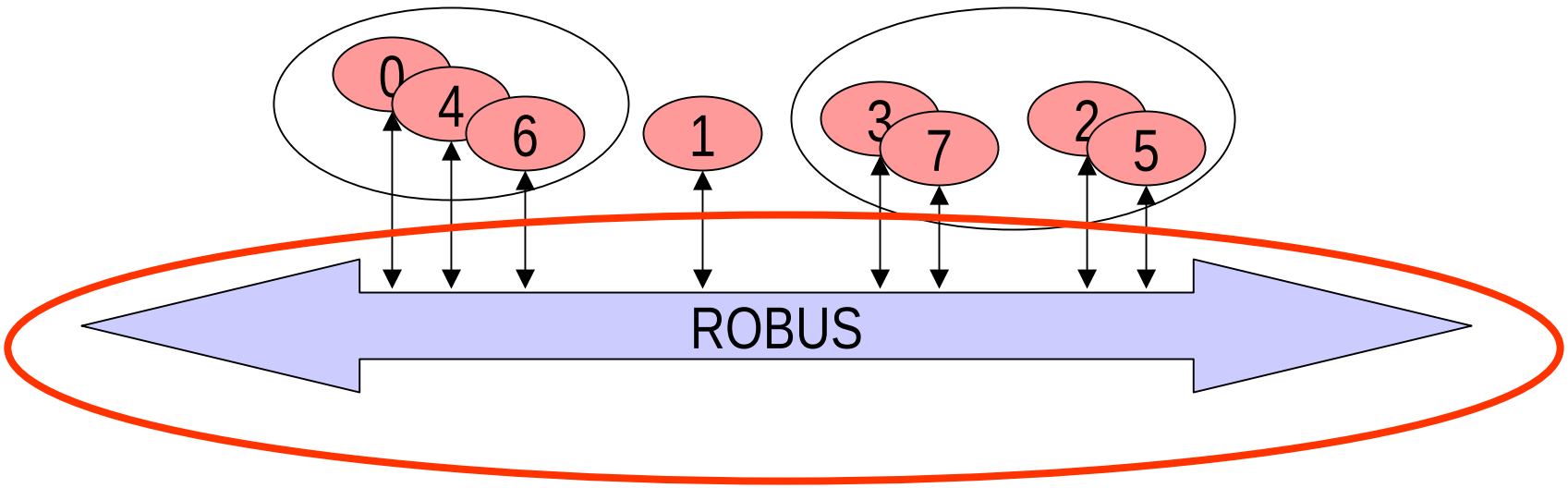
- Elemental Analysis – Appendix B, 3.3.1
- ~~Safety Specific Analysis – Appendix B, 3.3.2~~
- Formal Methods – Appendix B, 3.3.3

Design/Verification Hierarchy

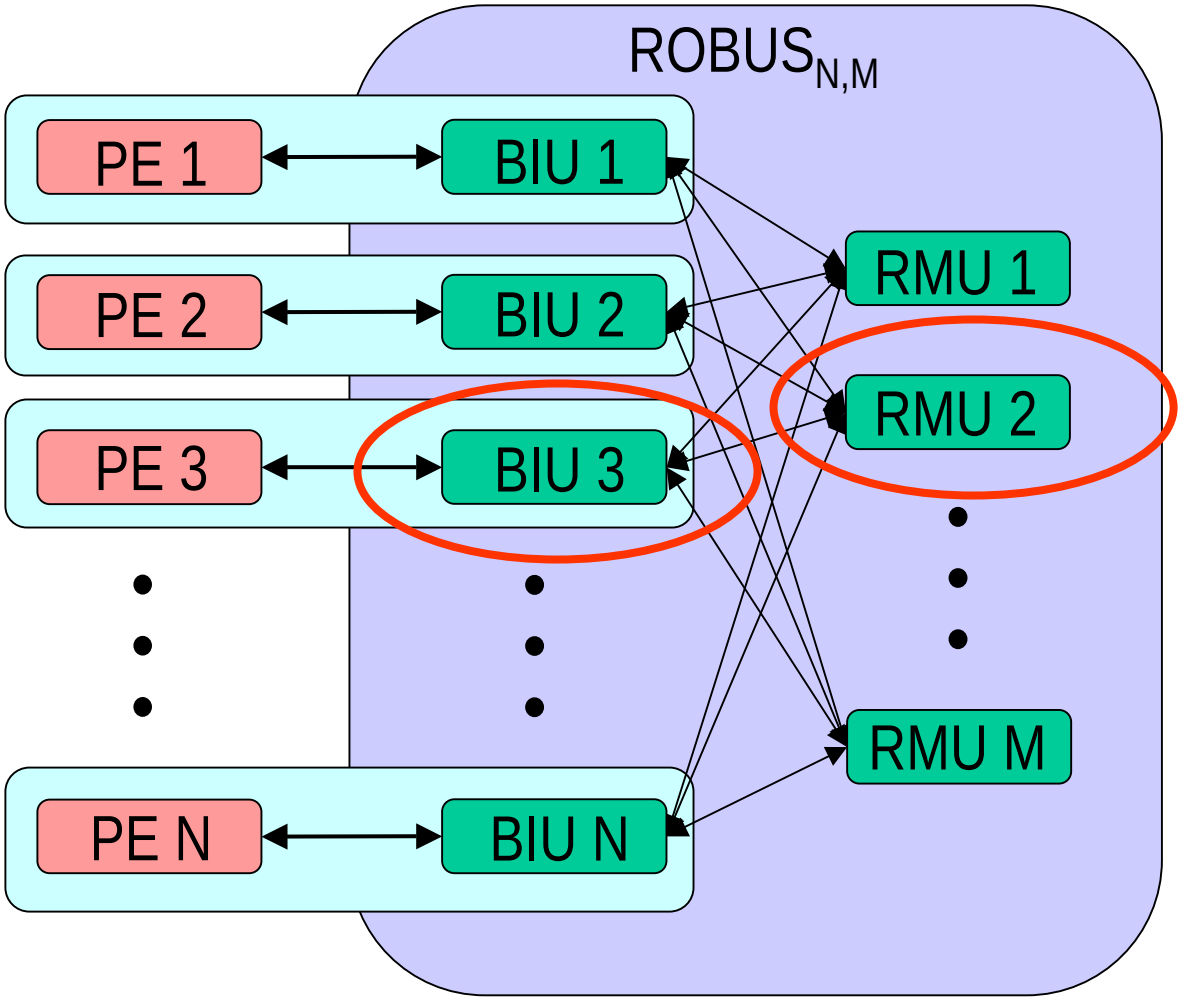


SPIDER

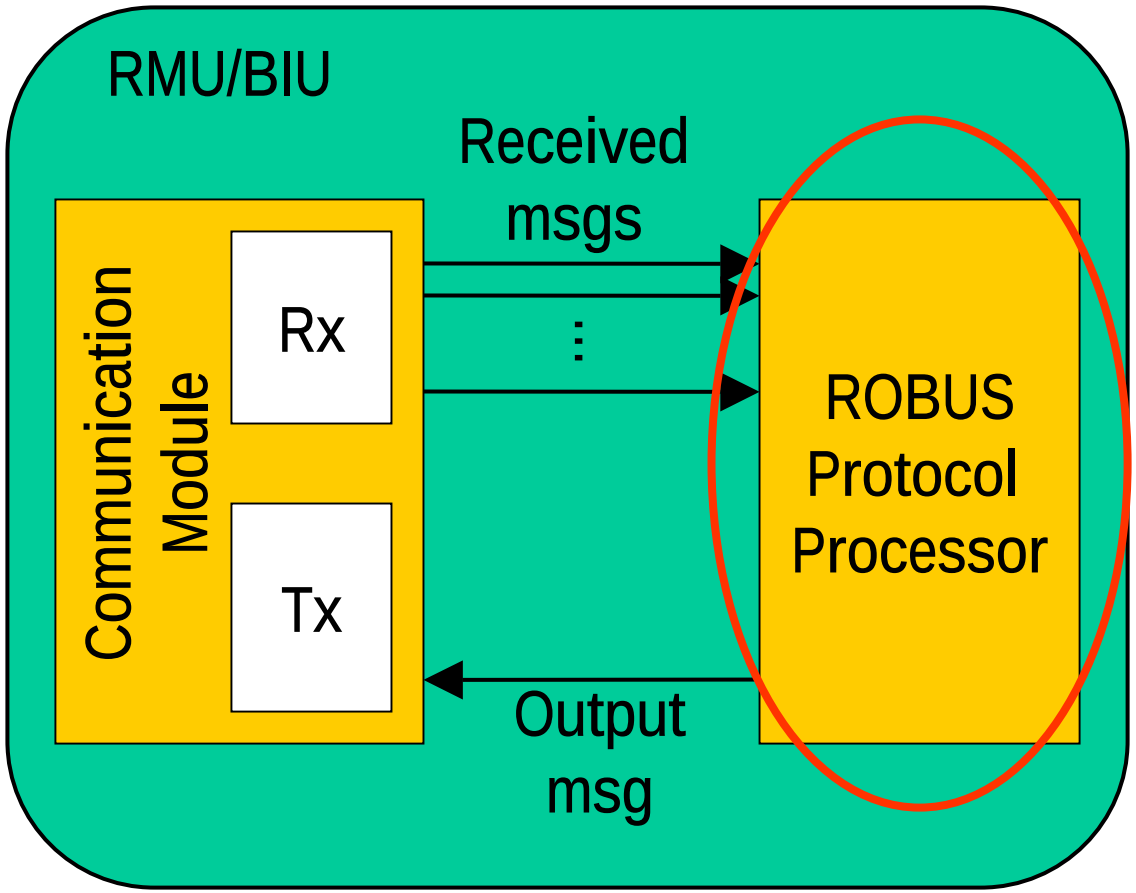
Sample SPIDER Configuration



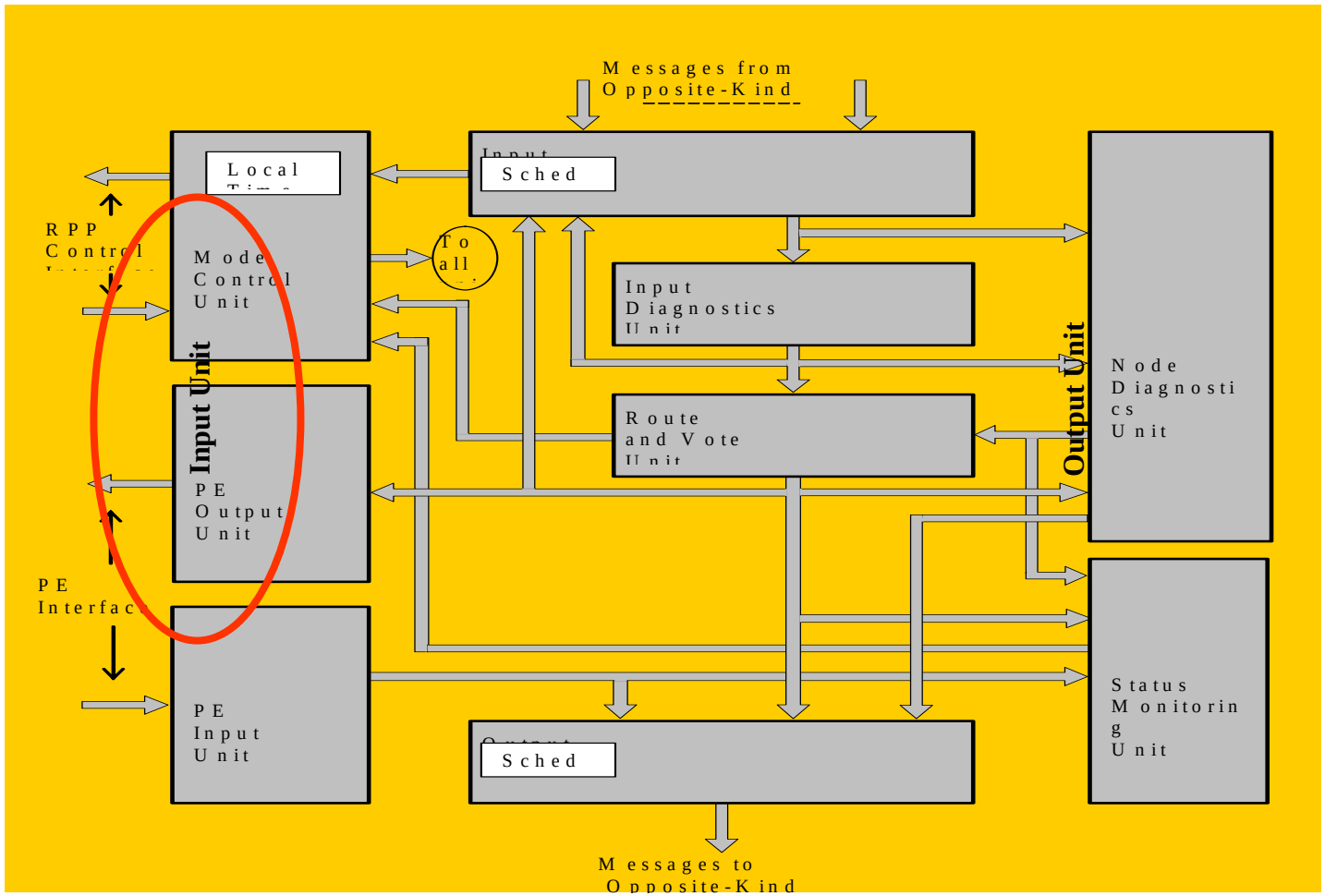
ROBUS Topology



RMU or BIU


RMU/BIU
Received msgs
Communication Module
Rx
Tx
ROBUS Protocol Processor
Output msg

ROBUS Protocol Processor (RPP)



Why focus on the Input Unit (IU)?

- IU includes both synchronous and asynchronous features
  - Typical of communication hardware
  - IU provides synchrony for other RPP units which allows other units to have a much more simple design
- IU is critical to RPP operation
  - IU is in the Functional Failure Path for all functions
- IU is the single largest unit in the RPP
  - IU is 4000 lines of VHDL
  - Using lines of VHDL metric, IU is 29% of RPP
  - Using synthesis area metric, IU is 27% of RPP

Elemental Analysis

- Analogous to structural coverage in software
  - A coverage criteria answers: how much testing?
  - DO-254 doesn't specify a criteria; so, which criteria should be used?
- Analysis is focused on the VHDL source for Input Unit
- For a full analysis
  - whole ROBUS Protocol Processor must be verified
  - COTS elements must be evaluated

Aspects of Elemental Analysis

- Functional Failure Path identification
  - All primary functions use the input unit
- Coverage criteria identification
  - “Focused Expression Coverage”
- Test environment
  - Test of simulation
  - Prototype tests are under consideration

Focused Expression Coverage (FEC)

- VN-cover's default condition coverage for VHDL is FEC
- VN-cover is from TransEDA*
- Why choose FEC?
  - We have NOT done research on appropriate coverage criteria for hardware
  - FEC is equivalent to masking MCDC
  - FAA has accepted masking MCDC for software projects

*NASA does not recommend any particular coverage tool

Equivalence of FEC and masking MCDC

- We have determined that FEC is equivalent to masking MC/DC
  - By two independent examinations of the TransEDA documentation
  - By hand comparison of results for simple designs
- From TransEDA documentation
  - "[FEC ensures] the output has been sensitized to the input, and the input has taken both possible values."
  - A pair of tests is defined as two tests where *only* the input being tested takes different values.
  - "Pairing of tests … is not a requirement because the requirement for full testability is simply that each input has taken both possible values while it is controlling and output."

Assessment of VN-cover

- DO-254 does not require detailed assessment of tools supporting elemental analysis
  - *"If the tool is … used to assess the completion of verification testing, such as in elemental analysis, no further assessment is necessary"* p. 78, item 4
- Really? … according to section 11.4, for level A and B functions the rational is given
  - An error in a design tool can introduce an error in the product (single point failure), therefore these tools require a "design tool qualification."
  - An error in a verification tool can allow the propagation of an error in the product (two failures), therefore less assessment is needed. These tools need a "basic test qualification."
  - Perhaps: An error in a coverage tool, could allow the propagation of an error in the testing process, which could allow the propagation of an error in the product (three failures), therefore an lower standard is required.

Elemental Analysis Status

- Completed ROBUS redesign
- Currently generating requirements-based test cases
- Haven't started testing
  - Preliminary investigations did not produce any surprises
- Recognized need to modify the Input Unit
  - Tool restricts use of VHDL language features
  - We should have been aware of this up front

Formal Methods

- Formal proof of key fault-tolerance protocols
  - Interactive Consistency
  - Distributed Diagnosis
  - Clock Synchronization
  - Restart
- Formalizing low-level requirements
  - investigating these for test-case generation

Summary

- Using ROBUS to explore advanced verification in Appendix B of DO-254
- Using “overlapping combinations” (elemental and formal) to provide design assurance
- “Focused Expression Coverage” is equivalent to masking MCDC
- For much more information see:

  http://shemesh.larc.nasa.gov/fm/spider/spider_pubs.html